

## Sistem Rekomendasi *Peer Helper* pada Forum Pembelajaran dengan Metode *K Nearest Neighbor*

Aditiyan Iswahyudin<sup>1</sup>, Dade Nurjanah, Ph.D<sup>2</sup>

<sup>1,2</sup>Fakultas Informatika, Universitas Telkom, Bandung

<sup>1</sup>aditiyan@students.telkomuniversity.ac.id, <sup>2</sup>dadenurjanah@telkomuniversity.ac.id

---

### Abstrak

Menurut teori pembelajaran Vygotsky, model pembelajaran kooperatif (atau bisa dibilang juga *peer learning*) adalah penekanan pembelajaran dengan model dialog yang dilakukan secara interaktif berbasis sosial. Keterlibatan *learner* lain membuka kesempatan bagi *learner* untuk mengevaluasi dan memperbaiki pemahaman dalam proses belajar. *Peer helper* merupakan *learner* yang mempunyai pengetahuan lebih yang mampu dan bersedia untuk membagikan ilmu dan pengalamannya kepada *learner* lain. *Peer learning* ini juga dapat dilakukan secara *online*, melalui suatu forum diskusi layaknya MOOC. Namun untuk menentukan siapa yang layak menjadi kandidat *peer helper* masih menjadi masalah utama. Tugas akhir ini menjelaskan tentang sebuah sistem untuk mengklasifikasi *learner* yang cocok menjadi kandidat *peer helper* dengan menggunakan metode KNN. *Peer helper* yang direkomendasikan diperoleh berdasarkan hasil klasifikasi peserta didik yang memiliki label untuk direkomendasikan menjadi *peer helper*. Uji coba menggunakan data hasil aktivitas *learner* dalam forum pembelajaran *online* dan dari data tersebut diterapkan metode KNN untuk mengklasifikasi *learner* yang cocok menjadi *peer helper*. Hasil dari uji coba didapat bahwa penggunaan KNN meningkatkan akurasi klasifikasi sebesar 0.20% dengan hasil 85.71%.

**Kata kunci :** *Recommender System, Peer Helper, KNN, Naïve Bayes, forum diskusi online*

---

### Abstract

According Vygotsky, cooperate learning model (also known as *peer learning*) emphasise learning using dialog model that happen interactively based on social interaction. Learner's involvement in the forum can open opportunities for other learner to evaluate and improve their understanding on learning process. Peer helper is a student who has more knowledge and willing to share his knowledge and experience with other students in a learning forum. Peer helper must be able to master and understand the subjects that are being studied well and have the desire to help other fellow students. However, to determine who is eligible to be a peer helper candidate is still a major problem. Therefore, due to this problem, this final project focuses on classifying students who are suitable to be peer helper candidates using the KNN method. Experiments are conducted using activity data from online learning forum. The experiments prove that KNN can improve the accuracy of classification by 0.20% in comparison with Naïve Bayes implemented in a previous study. The highest accuracy is 85.71%.

**Keywords:** *Recommender System, Peer Helper, KNN, Naïve Bayes, Online discussion forum*

---

### 1. Pendahuluan

#### Latar Belakang

Dalam suatu pembelajaran baik itu pembelajaran konvensional maupun pembelajaran berbasis teknologi metode belajar secara *peer-to-peer* sudah lumrah digunakan [1]. Dalam pembelajaran secara *peer-to-peer* ini ada dua komponen utama yang saling berinteraksi, yaitu *peer helper* dan *peer learner*. Metode *peer-to-peer* ini mendorong para *learner* dalam suatu forum pembelajaran untuk saling belajar dan mengajar antar *learner* [2]. Masalah mulai timbul ketika menentukan siapa yang sanggup dan mampu menjadi *peer helper* dalam suatu forum pembelajaran. Maka dari permasalahan tersebut tercetuslah sebuah gagasan untuk membuat sebuah sistem yang bisa mengenali kandidat *peer helper* berdasarkan aktivitasnya pada sebuah forum pembelajaran [3] [4].

Dalam beberapa tahun terakhir, penelitian telah menunjukkan perkembangan yang cukup pesat dalam hal *peer learning*, khususnya pada suatu forum pembelajaran (MOOC) [1] [5]. Penelitian-penelitian tersebut umumnya membahas tentang pengaruh forum pembelajaran di dalam kegiatan belajar-mengajar [2] [6] hingga bagaimana menilai suatu hasil capaian pembelajaran di dalam forum pembelajaran [7].

Metode yang pernah digunakan adalah *Naïve Bayes* dengan menggunakan algoritma *Expectation-Maximization* (EM) [3]. Kelebihan metode ini adalah *Naïve Bayes* merupakan metode yang memperoleh hasil akurasi terbaik dibanding metode-metode lainnya untuk menganalisis peserta didik dalam suatu forum [7] [12], namun tentu metode ini memiliki kekurangan yaitu penggunaan metode *Naïve Bayes* dengan dataset yang kecil menyebabkan nilai *precision* dan *recall* yang rendah. Untuk parameterternya tergolong homogen, yang artinya metode KNN dan *Naïve Bayes* dalam klasifikasi dapat menggunakan parameter yang sama untuk mendapatkan

hasil, yaitu *Messages, Threads, Words, Sentences, Reads, Time, AvgScoreMsg, Centrality, Prestige*, dan *Replies* [8]. Fokus pengembangan penelitian ini adalah dengan menggunakan metode yang berbeda, yaitu KNN (*K-Nearest Neighbor*). KNN merupakan suatu metode klasifikasi *supervised*, yang berarti untuk mengklasifikasi data metode ini membutuhkan data latih (*train*) agar dapat memprediksi hasil data yang diujikan.

### Rumusan Permasalahan

Metode yang digunakan pada penelitian sebelumnya untuk mencari *peer helper* yaitu menggunakan *Naïve Bayes* [3], untuk penggunaan KNN sendiri dirasa meringankan *workload* pada *hardware* karena KNN termasuk algoritma yang *lightweight* [9] dan juga algoritma KNN proses operasinya lebih efisien dibandingkan algoritma serupa, seperti *Naïve Bayes* [10]. Dalam *peer learning* ini, KNN dapat membuat model prediksi dengan tingkat akurasi lebih tinggi dibandingkan menggunakan *Decision Tree* [14]. Pada penelitian sebelumnya juga telah dilakukan klasifikasi untuk bidang *learning* dan didapat hasil akurasi rata-rata diatas 80% [18]. Dari pertimbangan-pertimbangan inilah maka metode *Naïve Bayes* perlu dibandingkan dengan metode lain sehingga didapatkan data perbandingan yang valid. Dari permasalahan tersebut, rumusan masalah pada penelitian ini adalah sebagai berikut :

Bagaimanakah performansi rekomen *peer learner* dengan menggunakan metode KNN dan bagaimanakah perbandingannya dengan metode *Naïve Bayes*?

### Tujuan

Tujuan dari rumusan permasalahan yang ada pada penelitian ini adalah untuk merancang sebuah sistem yang mampu mengklasifikasi kandidat *peer helper* di antara *learner* menggunakan algoritma KNN dan membandingkannya dengan penelitian terdahulu.

### Organisasi Tulisan

Penulisan bab pertama membahas tentang latar belakang, rumusan masalah, serta tujuan dari penelitian ini. Selanjutnya pada bab kedua dilanjutkan dengan membahas mengenai studi terkait yang berisi pengamatan terhadap paper-paper penelitian terdahulu baik *paper* yang diterbitkan maupun lewat buku dengan topik yang terkait dengan permasalahan yang ingin diangkat dalam penelitian ini. Pada bab ketiga dilakukan perancangan sistem menggunakan metode yang akan digunakan. Perancangan ini juga dilanjutkan dengan implementasi menggunakan data asli. Pada bab 4 dijelaskan hasil dari implementasi akan dianalisis dan dibandingkan dengan penelitian serupa yang sudah pernah dilakukan. Terakhir pada bab 5 membahas kesimpulan dari penelitian ini juga saran yang diberikan terkait penelitian.

## 2. Studi Terkait

### 2.1 Peer Learning

Dalam sebuah lingkungan kegiatan belajar mengajar seringkali *learner* membutuhkan persepsi dari *learner* yang tingkatnya di atas mereka [11] maupun bagi *learner* seangkatan yang memiliki pemahaman yang lebih [3] sebagai pemandu dalam belajar. Pengalaman dan ilmu inilah yang para *learner* butuhkan saat mereka belajar, baik dalam kelas secara *synchronous* maupun *asynchronous* layaknya pada sebuah *forum online*. *Peer Learning* ini bisa dideskripsikan sebagai interaksi sosial antara individu dengan individu, individu dengan kelompok, maupun kelompok dengan kelompok ini menghasilkan keuntungan secara dua arah [11]. Menurut Barnett (2008), interaksi seperti ini dapat mendobrak hambatan yang biasanya ditemui dan turut membantu dalam perkembangan kompetensi akademik maupun sosial. Andrews dan Clark (2011) mencetuskan beberapa model *peer learning*, yaitu lingkungan belajar individu dengan individu maupun antar kelompok dengan level ilmu yang sama ataupun level ilmu yang lebih tinggi untuk mentornya. Semua *learner* dapat memilih model yang diinginkan atau model tersebut langsung ditargetkan untuk *learner* yang dirasa membutuhkan bimbingan dengan model tersebut.

### 2.2 Peer Helper

Dewasa ini, perkembangan teknologi telah berkembang cukup pesat khususnya di bidang pendidikan yang menyediakan pendekatan-pendekatan dan juga memfasilitasi sumber-sumber bantuan untuk individu-individu yang memerlukannya [3]. Sumber-sumber ini antara lain bisa berupa catatan kuliah, latihan-latihan, kuis, dan juga solusi dari soal-soal yang diberikan [12]. Keunggulan-keunggulan dari sumber-sumber ini antara lain [3]:

1. Mempromosikan sosialisasi peserta dalam konteks kerja dan meningkatkan motivasi mereka dengan memberi kejujuran atas pengetahuan serta kelebihan dari peserta didik tersebut dalam hal yang positif.
2. Bantuan rekan sangat terletak dalam konteks bersama, oleh karena itu hal tersebut dapat memberikan pengalaman belajar yang lebih kuat bagi orang yang meminta bantuan.

3. Cara untuk membuat pembelajaran terjadi tepat pada waktunya, yaitu saat masalah muncul.
4. Mempromosikan proses penjelasan diri dan refleksi dalam helper, dan dengan cara ini pembelajaran timbal balik akan terjadi.
5. Efektif biaya, karena menggunakan peserta didik sebagai sumber pengetahuan sekaligus sebagai sumber pengajaran.

Penentuan *peer helper* yang sesuai dengan kriteria adalah dengan menggunakan data yang memuat aktivitas dan interaksi *learder* dengan *learner* lainnya yang ada pada forum pembelajaran. Data tersebut terdiri dari informasi pertanyaan-pertanyaan yang dapat dijawab oleh seorang individu yang ditanyakan *learner* dalam sebuah forum. Jawaban-jawaban tersebut diberikan *rating* tergantung *learner*, apakah jawaban tersebut solutif dan bermanfaat atau tidak. Semakin banyak dan tinggi *rating*-nya, semakin tinggi juga nilai individu tersebut menjadi kandidat *peer helper*. Untuk menilai interaksi sosial antar *learner* dinilai dari keaktifan yang mereka lakukan dalam forum. Keaktifan tersebut meliputi jumlah pesan (*posting*) yang dikirim, jumlah kata tiap pesannya, nilai *centrality* yaitu ketika *learner* menulis tanggapan terhadap *posting learner* lain, dan nilai *prestige* yaitu ketika *learner* mendapatkan tanggapan akan *posting*-an nya dari *learner* lain dalam sebuah forum [3]. Secara general, faktor-faktor dalam penentuan layak tidaknya seseorang sebagai *peer helper* dalam suatu forum pembelajaran adalah sebagai berikut [17]:

#### 1. *Sample and Data Collection*

Biasanya dilakukan dengan menggunakan kuesioner dengan topik pertanyaan yang mengukur seberapa menguntungkan sistem pembelajaran *peer helper* ini baik bagi *helper* maupun *learner*.

#### 2. *Measures*

Dalam pengukuran kuesioner terdapat beberapa variabel seperti :

##### A. *Self-Awareness*

Variabel ini mengukur tingkat kesadaran diri dari *peer helper* seperti memahami emosi negatif diri sendiri dan juga sadar akan kebutuhan untuk menjadi lebih baik untuk diri sendiri dan juga orang lain.

##### B. *Emotional Awareness of Others*

Variabel ini mengukur tingkat empati dan perhatian terhadap orang lain seperti kemampuan mendengar dan kepekaan terhadap orang lain.

##### C. *Organizational-Based Self-Esteem*

Variabel ini mengukur tingkat interaksi terhadap orang lain seperti seberapa nyaman anda dengan orang lain dan juga seberapa percaya diri anda apabila ada orang yang ingin meminta bantuan kepada anda.

#### 3. *Analytical Method*

Merupakan metode lanjut dalam pengukuran, yaitu membandingkan hasil sebelum dilakukannya *peer learning* dan sesudah dilakukannya *peer learning* baik itu *helper* maupun *learner*.

#### 4. *Skills Acquisition*

Pada bagian ini dilakukan rangkuman dari *skill* yang berguna dan dimiliki oleh *peer helper* seperti mampu merefleksikan hal yang dirasa secara jelas, mampu menjawab pertanyaan dan aktif menstimulasi *learner* dengan pertanyaan sesuai konteks, memberi perhatian, dan menerima dan memperbaiki kesalahan, mampu mengontrol emosi dan stress, mampu menyelesaikan konflik, dan mampu bekerja dalam tim.

### 2.3 *Peer Review and Reputation Metrics*

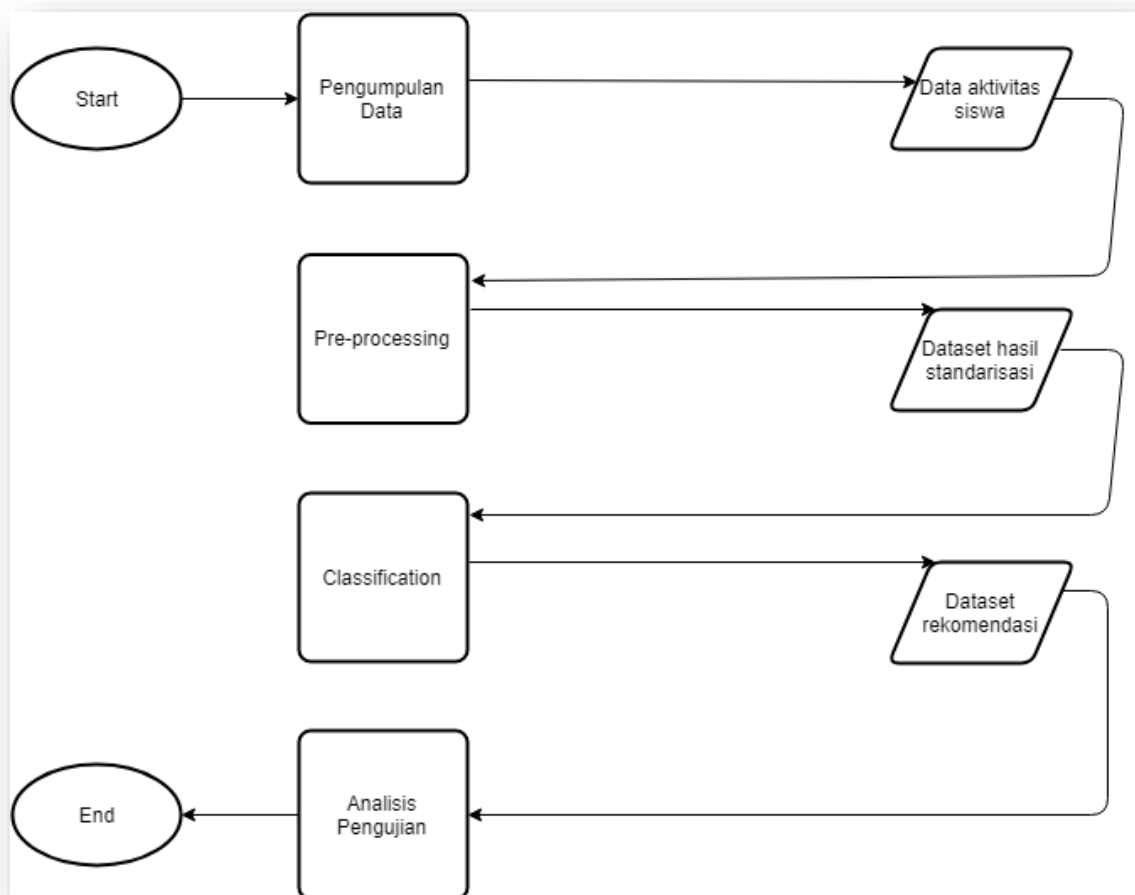
*Peer Review* adalah sebuah metode yang berguna untuk mengembangkan proses belajar-mengajar *learner* dengan cara me-review pembelajaran tersebut oleh *learner* lain [15]. Ada beberapa *tools* untuk membantu dalam proses *review*, Expertiza adalah salah satunya [16]. Expertiza merupakan sebuah *learning web forum* yang di dalamnya *learner* dapat memilih tugas yang bisa mereka kerjakan dan hasil tugas tersebut nantinya bisa di-*review* oleh *learner* lain, dengan kata lain Expertiza menggunakan konsep *Peer Learning*. Dalam proses *review learner* dapat memberikan 2 jenis *feedback*, yaitu secara kuantitatif (skor) atau secara kualitatif. Untuk penilaian oleh instruktur masalah yang ingin ditekankan adalah dalam prediksi nilai berdasarkan hasil *reviewer* pada model yang dibangun. Untuk menjawab masalah tersebut digunakanlah *Reputation Metrics* yang menggunakan RMSE (*Root Mean Square Error*) untuk membandingkan nilai hasil prediksi dengan nilai sebenarnya [15]. RMSE ini mengukur keefektifan dari model yang dibangun, apabila nilai RMSE nya besar maka bisa dibilang prediksi dari model yang dibangun akan kurang efektif, sebaliknya apabila nilai RMSE nya kecil maka prediksi dari model akan lebih efektif.

### 2.4 *Aplikasi KNN Pada Klasifikasi dalam Bidang Learning*

KNN atau *K Nearest Neighbor* adalah sebuah metode klasifikasi yang sudah banyak digunakan dalam bidang *data mining* dan *machine learning* karena implementasinya yang tergolong sederhana dan performansinya yang baik [10]. Klasifikasi menggunakan KNN terbukti baik untuk performansinya dalam kasus yang memiliki data banyak dan *error rate*-nya hampir menyerupai apabila menggunakan *Bayes Optimization*. Pada penelitian sebelumnya oleh Amita Goel (2017) membandingkan SVM dengan KNN, pada eksperimen dengan data simulasi sebanyak 10.000 data, hasil yang didapat adalah untuk akurasi SVM sebesar 98.9% sedangkan untuk KNN sendiri mendapat nilai akurasi sebesar 96.47% [13]. Dari hasil ini membuktikan walaupun SVM mendapat nilai akurasi lebih tinggi dibandingkan dengan KNN namun nilai akurasi KNN sendiri dirasa lebih baik dibandingkan dengan akurasi *Naïve Bayes* sebesar 85.51% [3]. Penggunaan KNN secara komputasional juga tidak terlalu intensif dan lebih mudah untuk diimplementasikan ketimbang SVM [13]. Hal inilah yang mendasari penelitian ini untuk membuktikan apakah benar KNN lebih baik daripada *Naïve Bayes* untuk kasus klasifikasi yang dibahas pada penelitian sebelumnya.

### 3. Sistem yang Dibangun

Rekomendasi *peer helper* akan menerapkan algoritma KNN untuk mendapatkan klasifikasi kandidat *peer helper* yang sesuai. Di bawah ini adalah gambaran umum dari sistem yang ingin dibangun.



Gambar 1. Gambaran Umum Sistem dan Alurnya

**Algorithm 1.** Classification KNN

Data: Dataset

Store: list

```

1: split Data into test and train
2: x_train, x_test, y_train, y_test
3: for each point in train Data do
4:   find Euclidean_Distance
5:   Store → Euclidean_Distance
6:   K : 0
7:   while K < 20 do
8:     K = K+1
9:     fit x_train with y_train
10:    predict x_test
11:    calculate and show RMSE value for optimal k
12:    show predicted value
13:   end while
14: end for

```

**Gambar 2.** Alur Klasifikasi KNN

Di mana  $x_{train}$ ,  $x_{test}$ ,  $y_{train}$ , dan  $y_{test}$  adalah hasil fungsi *split* pada *Python* yang berguna untuk memisahkan satu dataset menjadi data latih dan data uji untuk menguji keakuratan dari model klasifikasi yang dibangun. Cara *split* ini adalah pertama menentukan ukuran *train*, baik  $x_{train}$  dan  $y_{train}$ . Pada percobaan ini digunakan *split ratio* 0.1 yang maksudnya adalah 10% dari data akan digunakan sebagai *training set*. *Split* dengan cara memotong array dari *dataset* secara berulang menggunakan index dari array tersebut sebagai acuannya.  $x_{train}$  dan  $y_{train}$  digunakan untuk melatih dataset dengan menerapkan *Supervised Machine Learning* sehingga dapat menghindari *Overfitting* yang nantinya akan diujikan ke data  $x_{test}$  dan  $y_{test}$ .  $K$  adalah nilai yang akan digunakan dalam klasifikasi KNN, pada alur di atas terdapat nilai  $K$  sebanyak 20 untuk diuji akurasi menggunakan metode penghitungan RMSE. Nilai  $K$  dipilih sebanyak 20 karena saat eksperimen dilakukan nilai  $K$  diatas 20 nilai akurasi sudah stabil yang berarti nilai akurasi sudah tidak berbeda jauh antar  $K$  nya.

**3.1 Pengumpulan Data**

Forum pembelajaran berfungsi sebagai sarana instruktur dan *learner* berdiskusi dan bertukar pengetahuan sekaligus mendapatkan data guna penelitian ini. Pengumpulan data dilakukan dari hasil aktivitas *learner* didalam forum pembelajaran yang telah dibuat oleh instruktur yang berisikan materi yang telah ditentukan juga oleh instruktur tersebut. Data hasil aktivitas antara lain *messages*, *words*, *avgscoremsg*, *centrality*, dan *prestige*. *Output* dari pengumpulan data ini adalah dataset yang siap untuk di *pre-processing*. Data yang digunakan dalam penelitian ini adalah menggunakan dataset hasil aktivitas *learner* dalam forum pembelajaran IDEA Universitas Telkom dari penelitian sebelumnya [3].

**3.1.1 Data Hasil Aktivitas Learner**

Data-data yang disimpan dalam sistem disesuaikan dengan kebutuhan penelitian. Data-data tersebut difokuskan ke dalam tiap-tiap *thread* sehingga *peer helper* akan ditampilkan pada tiap forum. Pada penelitian ini forum yang digunakan adalah forum pembelajaran Telkom IDEA dengan focus di mata kuliah Algoritma Struktur Data. Data yang disimpan dalam forum antara lain:

**1. Messages**

*Messages* dalam sistem ini berupa *posting* atau komentar. Jumlah *post* tiap *learner* dalam *thread* akan dihitung. *Post* ini dapat berupa komentar akan sebuah *thread* atau komentar terhadap *post learner* lain. Apabila *learner* mem-post pada *thread*-nya sendiri atau mengomentari *post*-nya sendiri maka *message* tidak akan bertambah.

**2. Words**

Pada sistem ini, tiap kalimat yang ditulis oleh *learner* diberikan nilai 1.

### 3. AvgScoreMsg

Rata-rata dari *rating* akan dihitung dan disimpan sebagai parameter *AvgScoreMsg*. Nilai rata-rata ini mempunyai rentang dari 0 hingga 3. Nilai boleh bilangan bulat ataupun *real*. Penghitungan untuk *AvgScoreMsg* menggunakan rumus berikut:

$$avgscoremsg(A) = \frac{\text{nilai rating 1} + \text{nilai rating 2} + \dots + \text{nilai rating } n}{n}$$

Dimana  $n$  adalah jumlah *rating* yang diberikan oleh *learner*.

### 4. Centrality

*Centrality* mempunyai nilai 0 dan 1 di mana nilai ini menunjukkan rasio *out-link learner*. *Out-link* sendiri direpresentasikan dengan *learner* menulis tanggapan dari *learner* lain. Nilai boleh bilangan bulat ataupun *real*. Penghitungan untuk *centrality* menggunakan rumus berikut :

$$C(i) = \frac{d_o(i)}{n-1}$$

Dimana  $d_o(i)$  adalah jumlah dari *out-link* dan  $n$  merupakan jumlah *learner*. Dalam penelitian ini, jumlah *messages* merupakan *out-link* untuk menghitung *centrality*, karena pesan yang dikirim pada tiap *thread* dalam forum merupakan tanggapan dari *learner* terhadap *learner* lain.

### 5. Prestige

*Prestige* mempunyai nilai 0 dan 1 di mana nilai ini menunjukkan rasio *in-link learner*. *In-link* sendiri direpresentasikan dengan *learner* menerima tanggapan dari *learner* lain. Nilai boleh bilangan bulat ataupun *real*. Penghitungan untuk *prestige* menggunakan rumus berikut :

$$P(i) = \frac{d_o(i)}{n-1}$$

Dimana  $d_o(i)$  adalah jumlah dari *in-link* dan  $n$  merupakan jumlah *learner*. *In-link* dapat diketahui dari jumlah tanggapan dari *learner* lain terhadap *messages (posting)* yang telah diberikan dalam *thread* suatu forum.

**Tabel 2. Contoh data hasil aktivitas *learner***

Nama	Messages	SumWord	AvgScoreMsg	Centrality	Prestige
Mahasiswa_1	6	143	2.73	0.098	0.246
Mahasiswa_2	1	24	1.75	0.016	0.033
Mahasiswa_3	1	12	0	0.016	0.015
Mahasiswa_4	4	68	2	0.066	0.116
Mahasiswa_5	2	69	2	0.033	0

## 3.2 Pre-processing

Dataset hasil pengumpulan data lalu di *pre-processing* yaitu dengan cara *Transforming Standardization* [13]. *Transforming Standardization* adalah sebuah proses mengubah nilai data sehingga data tersebut mempunyai *mean* = 0 dan standar deviasi = 1. Standar deviasi merupakan suatu cara ukur untuk menentukan jumlah variasi atau *disperse* nilai dari sebuah dataset. Mean merupakan rata-rata nilai dari dataset. Contoh data yang sudah di proses bisa dilihat pada Tabel 3.

**Tabel 3. Hasil *Standardize***

Nama	Messages	SumWord	AvgScoreMsg	Centrality	Prestige
------	----------	---------	-------------	------------	----------

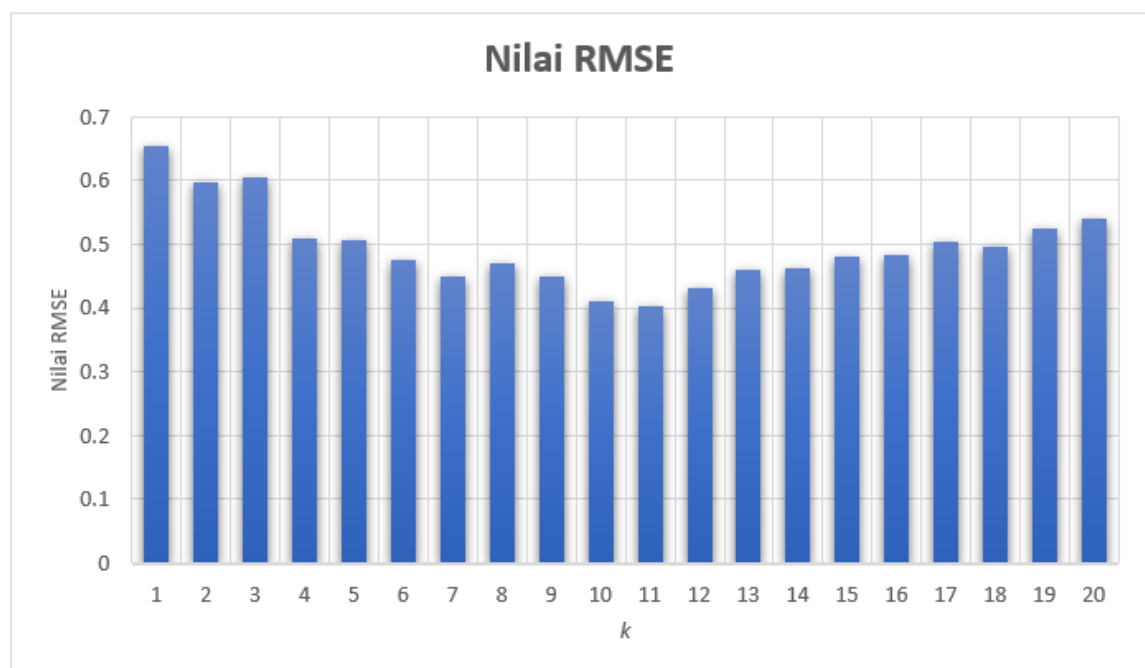
Mahasiswa_1	2.870762	2.067052	1.656209	2.921570	4.355465
Mahasiswa_2	-0.513026	-0.186303	0.687576	-0.508745	0.007244
Mahasiswa_3	-0.513026	-0.413533	-1.042125	-0.508745	-0.360212
Mahasiswa_4	1.517247	0.646870	0.934676	1.582911	1.701621
Mahasiswa_5	0.163732	0.665806	0.934676	0.202418	-0.666424

Hasil dari *preprocessing* ini adalah dataset yang siap untuk klasifikasi.

#### 4. Evaluasi

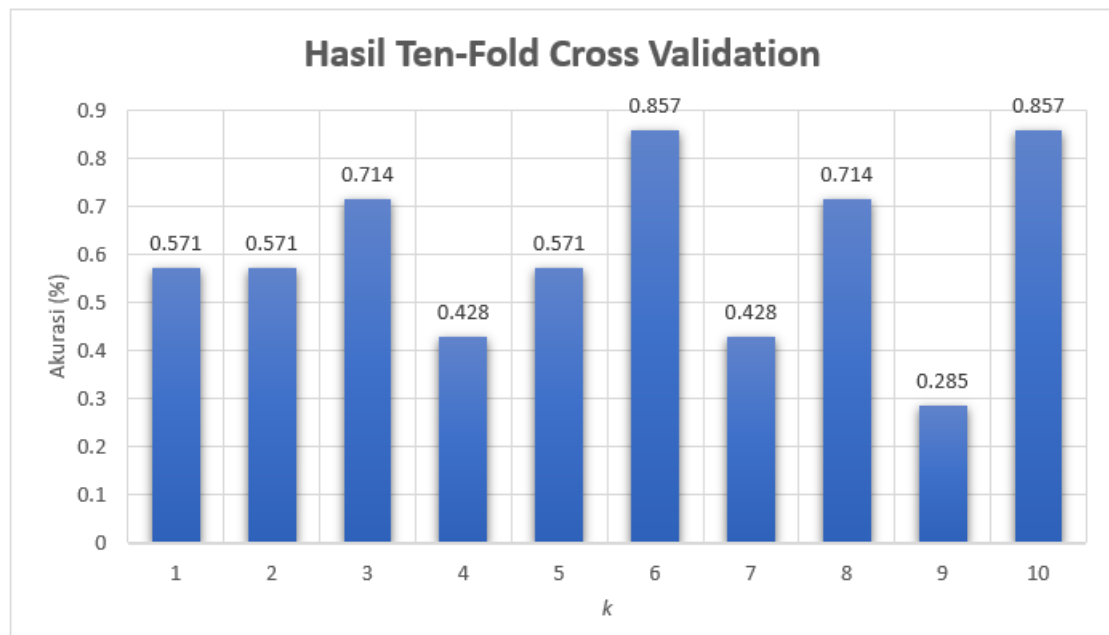
##### 4.1 Hasil Pengujian

Dari pengujian yang telah dilakukan didapatkan dataset klasifikasi yang akan mengolah data latih untuk memprediksi label dari sekumpulan data test yang belum diketahui kategori labelnya. Klasifikasi di sini menggunakan metode KNN untuk mendapatkan hasil rekomendasi. Adapun tahapan dalam proses klasifikasi ini dibagi menjadi 3, yaitu *men-split* dataset *train*, melatih data *train*, lalu menguji dataset menggunakan *baseline* dari data *train*. Pada model ini dalam penentuan nilai  $k$  optimal dilihat dari nilai RMSE (*Root Mean Square Error*) yang paling kecil dari tiap iterasi nilai  $k$  dari 1 sampai 20. Nilai dari tiap iterasi bisa dilihat pada Gambar 4.



Gambar 4. Nilai RMSE

Bisa dilihat dari Gambar 4 bahwa nilai  $k$  yang mempunyai nilai RMSE terkecil adalah 11, yang berarti  $k$  11 merupakan nilai  $k$  yang optimal dan karena mempunyai nilai *error* yang rendah. Sebagai perbandingan, digunakan juga metode *10-Fold Cross Validation* dalam penentuan *split* data test dan data train. Dari 10 kali *fold* didapat lah nilai sebagai berikut:



**Gambar 5. Hasil Ten-Fold Cross Validation**

Pada Tabel 5 terdapat 3 hasil *fold* dengan hasil akurasi terendah yaitu *fold* ke 4, 7, dan 9. Pada *fold cross validation* data dibagi pada sebuah segmen/blok sebanyak  $k$ . Pada kasus di atas,  $k = 10$  maka dataset akan dibagi sebanyak 10 blok secara acak. Dari 10 blok tersebut, terdapat blok yang akan digunakan sebagai data training dan juga data test, yang di mana data test tersebut merupakan *fold* ke- $k$ . Maka, data pada blok 4, 7, dan 9 yang digunakan sebagai data test mempunyai nilai akurasi rendah karena nilai dari data yang ada pada blok-blok tersebut mempunyai *variance* rendah dengan *bias* tinggi yang menimbulkan *underfitting* sehingga data tidak diprediksi secara akurat (atau hasil prediksinya buruk).

Setelah didapat hasil rekomendasi menggunakan algoritma KNN lalu hasilnya dibandingkan dengan hasil rekomendasi yang didapat menggunakan Naïve Bayes. Kedua hasil rekomendasi tersebut lalu dibandingkan akurasinya dan dianalisis mana dari kedua algoritma tersebut yang mempunyai akurasi terbaik.

Dari hasil eksperimen yang dilakukan menggunakan metode KNN pada dataset didapat akurasi sebesar 85.71 % di mana dari 63 data, 54 diantaranya berhasil diprediksi secara benar sehingga menghasilkan 23 orang yang dikategorikan cocok sebagai *peer learner*. Proses klasifikasi yang digunakan menggunakan metode KNN dilakukan untuk mengklasifikasikan peserta didik yang dapat menjadi *peer helper* pada suatu forum diskusi dalam sistem pembelajaran *online* Universitas Telkom (IDE). Dengan menggunakan RMSE, akurasi yang dihasilkan dari tiap nilai  $k$  menghasilkan akurasi yang berbeda. Data yang didapatkan untuk menghitung pengujian nilai akurasi, didapatkan dari forum diskusi yang terdapat dalam web u-learning. Implementasi web u-learning yang dilakukan pada kelas IF41-04 dan IF-41-05 Universitas Telkom University pada mata kuliah Algoritma Struktur Data. Data ini digunakan agar hasil dari eksperimen menggunakan KNN dapat dibandingkan dengan penelitian sebelumnya yang menggunakan Naïve Bayes karena apabila menggunakan *dataset* lain maka hasilnya tidak bisa dibandingkan dengan penelitian sebelumnya. Pada eksperimen menggunakan KNN didapat akurasi sebesar 85.71 % dengan besar data *split* dan data *training* nya sama besarnya yang menghasilkan akurasi lebih besar ketimbang rata-rata akurasi apabila menggunakan metode Naïve Bayes yaitu sebesar 85.51 % yang di mana penggunaan KNN ini meningkatkan akurasi sebesar 0.20 %.

## 5. Kesimpulan

Berdasarkan hasil pengujian dan analisis yang telah dilakukan sebelumnya, maka bisa ditarik kesimpulan yaitu: Metode KNN mampu untuk meningkatkan akurasi klasifikasi sebesar 0.20% dari metode sebelumnya yaitu Naïve Bayes. Dari 63 mahasiswa, sistem dapat mengklasifikasi 23 mahasiswa yang cocok menjadi *peer helper*. Hasil akurasi yang didapat menggunakan KNN adalah sebesar 85.71% sedangkan pada Naïve Bayes sebesar 85.51%. Maka bisa disimpulkan bahwa performansi rekomen *peer helper* dengan menggunakan metode KNN meningkat.

Berdasarkan kesimpulan penelitian, maka ada beberapa saran untuk penelitian kedepannya seperti pada penelitian ini menggunakan dataset yang tergolong kecil yang hasilnya belum tentu akan sama apabila diaplikasikan pada dataset yang besar. Untuk metode KNN nya sendiri pada penelitian ini menggunakan dataset dengan data *split* yang tetap di mana apabila ada perubahan jumlah data maka nilai akurasi bisa berubah. Karena pada penelitian ini sudah didapat hasil dari pengujian menggunakan KNN, pada penelitian selanjutnya bisa



dilanjutkan dengan menggunakan metode SVM guna membandingkan hasilnya dengan KNN maupun Naïve Bayes.

#### Daftar Pustaka

- [1] G. Kennedy, C. Coffrin, P. de Barba, and L. Corrin, "Predicting success: how learners' prior knowledge, skills and activities predict MOOC performance," in *Proceedings of the Fifth International Conference on Learning Analytics And Knowledge - LAK '15*, Poughkeepsie, New York, 2015, pp. 136–140.
- [2] H. Labarthe, F. Bouchet, R. Bachelet, and K. Yacef, "Does a Peer Recommender Foster Students' Engagement in MOOCs?," p. 6.
- [3] A. Nugraha, "Rekomendasi Peer Helper pada Forum Ubiquitous Learning Menggunakan Naïve Bayes" p. 21.
- [4] K. Oku, S. Nakajima, J. Miyazaki, S. Uemura, and H. Kato, "A ranking method based on users' contexts for information recommendation," in *Proceedings of the 2nd international conference on Ubiquitous information management and communication - ICUIMC '08*, Suwon, Korea, 2008, p. 289.
- [5] F. Bouchet, H. Labarthe, K. Yacef, and R. Bachelet, "Comparing Peer Recommendation Strategies in a MOOC," in *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization - UMAP '17*, Bratislava, Slovakia, 2017, pp. 129–134.
- [6] J. Bobadilla, A. Hernando, and A. Arroyo, "e-learning experience using recommender systems," in *Proceedings of the 42nd ACM technical symposium on Computer science education - SIGCSE '11*, Dallas, TX, USA, 2011, p. 477.
- [7] M. I. López, J. M. Luna, C. Romero, and S. Ventura, "Classification via clustering for predicting final marks based on student participation in forums," p. 4.
- [8] A. Töschler, M. Jahrer, and R. Legenstein, "Improved neighborhood-based algorithms for large-scale recommender systems," in *Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition - NETFLIX '08*, Las Vegas, Nevada, 2008, pp. 1–6.
- [9] T. De Pessemier, K. Vanhecke, and L. Martens, "A scalable, high-performance Algorithm for hybrid job recommendations," in *Proceedings of the Recommender Systems Challenge on - RecSys Challenge '16*, Boston, Massachusetts, 2016, pp. 1–4.
- [10] S. Zhang, X. Li, M. Zong, X. Zhu, and D. Cheng, "Learning k for kNN Classification," *ACM Trans. Intell. Syst. Technol.*, vol. 8, no. 3, pp. 1–19, Jan. 2017.
- [11] J. Thalluri, J. A. O'Flaherty, and P. L. Shepherd, "Classmate peer-coaching: 'A Study Buddy Support scheme,'" p. 14.
- [12] C. Romero, M.-I. López, J.-M. Luna, and S. Ventura, "Predicting students' final performance from participation in on-line discussion forums," *Comput. Educ.*, vol. 68, pp. 458–472, Oct. 2013.
- [13] D. A. Goel and S. Mahajan, "Comparison: KNN & SVM Algorithm," *Eng. Technol.*, vol. 5, p. 4.
- [14] J. Andrews and R. Clark, "How Peer Mentoring Enhances Student Success in Higher Education," *Aston University*, Birmingham, 2011, p. 102.
- [15] D. Y. Lee, F. Pramudianto, E. F. Gehringer, "Prediction of Grades for Reviewing with Automated peer-review and Reputation Metrics," *North Carolina University.*, Raleigh, 2016.
- [16] Gehringer, E., "Expertiza: information management for collaborative learning." *Monitoring and Assessment in Online Collaborative Environments: Emergent Computational Technologies for E-Learning Support*, pp 143-159, 2009.
- [17] TAN, Yip Wei, Gilbert and HSI, Timothy. *The Impact of Peer-Helper Program on Peer Helpers: Some Preliminary Findings*. (2007). 8th International Conference on Human Resource Development Research and Practice across Europe, Oxford, 27-29 June 2007. 1-18. Research Collection Lee Kong Chian School Of Business
- [18] Alfere, Sameh & Maghari, Ashraf. (2018). *Prediction of Student's Performance Using Modified KNN Classifiers*. *UP Journal for Research and Studies*. 7. 134-142. .